Defence Research and Development Canada
Recherche et développement pour la défense Canada

Parallelization of an Urban Microscale Flow Model (urbanSTREAM)

Component 1 of CRTI Project 02-0093RD

Fue-Sang Lien and Kun-Jung Hsieh
Waterloo CFD Engineering Consulting, Inc.

Contract Scientific Authority: E. Yee, DRDC Suffield

Defence R&D Canada

Contract Report

DRDC Suffield CR 2008-024

December 2007

Canada

Parallelization of an Urban Microscale Flow Model (urbanSTREAM)

Component 1 of CRTI Project 02-0093RD

Fue-Sang Lien and Kun-Jung Hsieh
Waterloo CFD Engineering Consulting, Inc.
200 University Avenue West
Waterloo ON N2L 3G1

Contract Number: W7702-03R966

Contract Scientific Authority: E. Yee (403-544-4605)

Defence R&D Canada – Suffield

Contract Report

DRDC Suffield CR 2008-024

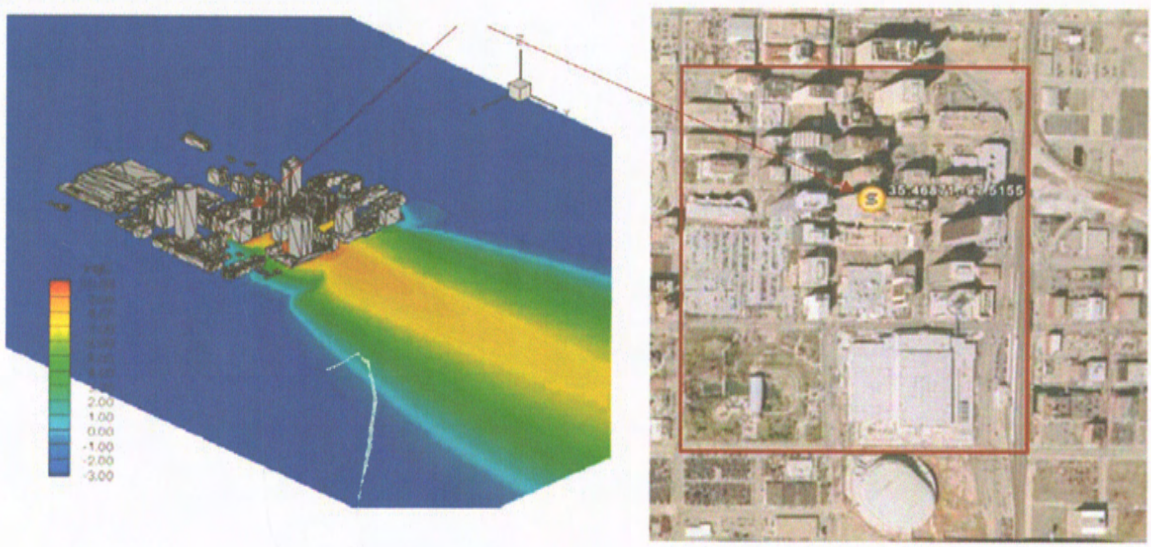
December 2007

Parallelization of an Urban Microscale Flow Model (urbanSTREAM): Component 1 of CRTI Project 02-093RD

Fue-Sang Lien and Kun-Jung Hsieh
Waterloo CFD Engineering Consulting, Inc.

WatCFD Report WD001/07

December 2007



ABSTRACT

The provision of an efficient and accurate Computational Fluid Dynamics (CFD) solver for simulating flows within an urban complex on the microscale (with a characteristic length scale extending to about 1-2 km) is required to support the development of an advanced atmospheric transport and diffusion multiscale modeling system for hazard assessment of chemical, biological, radiological or nuclear (CBRN) agents released within cities. The ultimate goal of this supporting effort is to construct a virtual test facility (VTF) that could eventually be used to develop guidelines and procedures for operating in the urban complex after a CBRN incident.

Details of the numerical algorithms used for the "stand-alone" microscale CFD solver urbanSTREAM executing on serial computers have been described in [1]. The objectives of this report are twofold. The first objective is to provide a description of the coupling of urbanSTREAM with an "urbanized" meso-gamma scale flow model (Global Environmental Multi-scale Local Area Model, or GEM LAM, developed by Canadian Meteorological Centre). A "one-way interaction" scheme is adopted, which allows for the matching of velocity and turbulence fields in any overlap region in a physically and mathematically consistent manner that preserves the physical conservation laws, mutually satisfies mathematical boundary conditions, and preserves numerical accuracy. The second objective is to provide implementation details about how urbanSTREAM is parallelized for use on PC clusters and the IBM massively parallel supercomputing platform to give a problem-solving environment for full 3-D parallel simulation. Validation of this fully-coupled mesoscale-to-microscale system is achieved by detailed comparison of model predictions with comprehensive, high-quality full-scale urban measurements in a real cityscape – the Joint Urban 2003 (JU2003) conducted in Oklahoma City in the US.

Table of Contents

ABSTRACT 2

1. INTRODUCTION 6

2. MULTI-BLOCK ALGORITHM 6

3. PORTING TO DISTRIBUTED-MEMORY MACHINE 8

4. APPLICATION TO AN ARRAY OF OBSTACLES 13

5. APPLICATION TO JOINT URBAN 2003 13

6. CONCLUSIONS 15

REFERENCES 18

List of Figures

Figure 1. Multi-block arrangement with different local coordinate systems. 19
Figure 2. All possible permutations of local coordinate systems in blocks adjacent to any reference block. 19
Figure 3. Solution sequence of SIMPLE algorithm within a multi-block scheme. 19
Figure 4. A multi-block mesh for a 2D two-element airfoil [7]. 20
Figure 5. Interfacing urbanGRID and urbanSTREAM-P. 20
Figure 6. Domain decomposition of a solution domain for 8 CPUs. 21
Figure 7. One-eighth of the computational mesh, including the "halo data" regions. 21
Figure 8. Illustration of data swapping between two adjacent halo data regions. 22
Figure 9 Illustration of data swapping by using MPI_SEND and MPI_RECV routines in halo data regions. 22
Figure 10 Index system used for the block in question and four neighboring blocks in a 2-D environment. 23
Figure 11. Flow over a regular and aligned array of obstacles obtained with URANS on a 4-CPU parallel system. 24
Figure 12. Profiles of streamwise velocity and turbulence kinetic energy (TKE) for a flow over a regular and aligned array of obstacles using URANS. Case 1: 2^{nd}-order time-stepping; Case 2: 1^{st}-order time stepping. 24
Figure 13. Parallel efficiency measured on the Nexus and Australis computing systems on Westgrid (www.westgrid.ca) for a flow over a regular and aligned array of obstacles using URANS. 25
Figure 14. Illustration of the present drag-force approach applied to the JU2003 test problem. 25
Figure 15. Flow visualization obtained with urbanSTREAM-P using power-law velocity profiles as inflow conditions for south-east, south-west, north-east and north-west wind directions in Oklahoma City. 26
Figure 16. Parallel efficiency measured on the Narwhal computing system on Sharcnet (www.sharcnet.ca) for a flow over Oklahoma City for south-west wind direction using URANS.. 26
Figure 17. Vertical Profiles of mean wind speed and wind direction at 35.46474° N and 97.51822° W using inflow conditions interpolated from the early results of the urban GEM LAM model for both serial and parallel versions of urbanSTREAM. 27
Figure 18. Comparison of vertical profiles of mean wind speed and wind direction interpolated from GEM LAM with experimental measurements at 35.45295° N and 97.52534° W, which is at 2 km south of the central business district of Oklahoma City. 27
Figure 19. Comparison of vertical profiles of mean wind speed and wind direction interpolated from GEM LAM with experimental measurements at 35.46474° N and 97.51822° W. 28
Figure 20. Comparison of vertical profiles of mean wind speed and wind direction interpolated from GEM LAM with experimental measurements at 35.47866° N and 97.51707° W. 28
Figure 21. Mean plume concentration isopleths (logarithmic scale) at ground level for tracer release on south side of Park Avenue in Oklahoma City. 29

Figure 22. Comparison between predicted mean concentrations obtained from urbanEU with flow fields provided by urbanSTREAM-P in standalone (PNNL) and coupled (GEM/LAM) modes, and experimental measurements obtained with detectors along Kerr Avenue and McGee Avenue. 29
Figure 23. Comparison between predicted mean concentrations obtained from urbanEU with flow fields provided by urbanSTREAM-P in standalone (PNNL) and coupled (GEM/LAM) modes, and experimental measurements obtained with detectors along 4th Street and 5th Street. 30
Figure 24. Comparison between predicted mean concentrations obtained from urbanEU with flow fields provided by urbanSTREAM-P in standalone (PNNL) and coupled (GEM/LAM) modes, and experimental measurements obtained with detectors along 6th Street and the 1-km sampling arc. 30
Figure 25. Flow chart of fully-coupled multigrid method for 2-D turbulent flows. 31
Figure 26. Mesh generation for local-grid-refinement with application to multiple source scenarios. 31

1. INTRODUCTION

The objective of CRTI Project 02-0093RD is to develop and validate a prototype state-of-the-science multiscale modeling system for predicting the transport and dispersion of chemical, biological, radiological or nuclear (CBRN) materials in the urban environment and beyond. This prototype modeling system serves as the basis of a high-fidelity predictive tool for scenario planning, forensic and post-event analysis, as well as for operational response. Incorporating the proposed system's capabilities in a government operations center will result in improved emergency preparedness for and management of potential CBRN incidents in Canadian cities. The tool can be used for planning at events of national significance (e.g., G8 summit, APEC meeting, 2010 Winter Olympics).

The details of the urban microscale modeling system developed for Component 1 of CRTI Project 02-0093RD, along with its five main modules (viz., urbanGRID, urbanSTREAM, urbanEU, urbanAEU and urbanPOST) are described in [1]. Results shown in [1] were obtained with the serial version of urbanSTREAM. In order to significantly reduce the wall-clock time and render the use of urbanSTREAM more feasible for emergency response, it is imperative that urbanSTREAM be implemented on a parallel computer system. This report will describe in detail how urbanSTREAM is parallelized on a distributed-memory computer system using the Message-Passing Interface (MPI) library [2], in conjunction with the "domain-decomposition" approach. In what follows, "urbanSTREAM-P" will be used to denote the parallel version of urbanSTREAM.

The concept underlying the multi-block algorithm will first be described in Section 2, including changes to the index system in the DO loop and construction of the "halo data" regions required to be incorporated into urbanSTREAM. Porting the multi-block version of urbanSTREAM to distributed-memory computer architectures using MPI will be addressed in Section 3. Validation of urbanSTREAM-P, including measurement of its parallel efficiency on various parallel systems, for two test problems involving the simulation of the disturbed flow field through an array of obstacles and Oklahoma City will be discussed in Sections 4 and 5, respectively. Finally, Section 6 outlines conclusions and recommendations for future extensions to the present study.

2. MULTI-BLOCK ALGORITHM

The multi-block algorithm [3] relies, essentially, on sub-dividing the solution domain into an arbitrary number of contiguous, non-overlapping blocks, each having its own grid and (if necessary) its own associated local coordinate system. Each grid is first generated separately by use of a suitable grid-generation procedure, the only constraint being continuity in grid-line positions across the block boundaries. Each block, viewed in isolation, is then surrounded by an 'auxiliary' layer of two cells originating from neighboring blocks. In reality, the block is made to penetrate into its neighbors to the extent of two cells in order to accommodate 'halo data', which are needed for the solution

within the block in question. The choice of a two-cell penetration is linked to the nature of the higher-order convection scheme and the Rhie & Chow interpolation practice [4]:

$$\begin{aligned}\bar{u}_e &= \frac{1}{2}(\bar{u}_P + \bar{u}_E) \\ &+ \frac{1}{2}\left[D_P^{\bar{u}}(\bar{p}_e - \bar{p}_w) + D_E^{\bar{u}}(\bar{p}_{ee} - \bar{p}_e) - \left(D_P^{\bar{u}} + D_E^{\bar{u}}\right)(\bar{p}_E - \bar{p}_P) \right]\end{aligned} \tag{1}$$

where, e.g., $D_P^{\bar{u}} = A_P / a_P$ (see Eqs. (40), (45) in [1]).

Although the coordinate systems of neighboring blocks can be quite different in orientation, as exemplified in Figure 1, all geometric data pertaining to the auxiliary layer attached to the parent block, including the metric tensors and the Jacobian, are treated in terms of the coordinate system of the parent block and are stored as if the layer were part of the block. This arrangement obviates, with one exception noted below, the need for any one block to directly access the 'foreign' geometric information and mass fluxes residing in the neighboring blocks during the solution process within the individual block.

Inter-block connectivity is handled using a connectivity matrix in the form of the 2D array MCONEC(BLOCK,FACE), where BLOCK is the block number being considered, FACE identifies the block face (ranging from 1 to 6, with 1 denoting the eastern face, 2 the western face, 3 the northern face, 4 the southern face, 5 the top face and 6 the bottom face), and MCONEC denotes the block sharing FACE with BLOCK. The coordinate system relating to any one block is stored in the form of COORD(BLOCK,FACE), representing all possible coordinate permutations in the neighboring block sharing the face 'FACE'. To convey the basic idea without introducing a significant loss of generality, a typical 2D example for COORD is illustrated in Figure 2. The neighboring right-hand-side block can take any one of eight combinations of coordinates, signified by the integers 1 to 8. Another example illustrating the use of both MCONEC and COORD is given below by reference to Figures 1 and 2, where

MCONEC(3,1)=2, MCONEC(3,2)=0,
MCONEC(3,3)=0, MCONEC(3,4)=1,

and

COORD (3,1)=8, COORD (3,2)=0,
COORD (3,3)=0, COORD (3,4)=1.

In the above, the value '0' signifies that the neighboring block is a physical (real) boundary of the solution domain.

By reference to Eq. (1) and the $a_E^{\bar{p}}$ coefficient in the pressure-correction equation for the SIMPLE algorithm [5] (see Eq. (49a) in [1]):

$$a_E^{\bar{p}} = D_e^{\bar{u}} A_e \tag{2}$$

where $\bar{D^{u}}A$ at cell face 'e' are obtained as a centered average of the values of $\bar{D^{u}}A$ at the two neighboring nodes on either side of the face (viz., at nodes P and E), and the coefficients in the pressure-correction equation depend on the a_P value associated with the momentum equations applied to the cell over which mass conservation is to be satisfied, as well as two neighboring cells on either side in any coordinate direction. It is crucial, therefore, to transmit this quantity from the neighboring blocks into the auxiliary two-cell layer when solving the momentum equation in the parent block. This transfer is greatly assisted by the fact that a_P is coordinate-invariant, i.e., independent of the block-local coordinate system.

Once the coefficients for the transport and pressure-correction equations have been assembled for each block, the resulting system of equations is solved in a segregated manner as illustrated in Figure 3. Each set of equations pertaining to any one block is solved within an 'inner iteration' by Stone's SIP method [6] concurrently with the temporarily 'frozen' block boundary conditions in the 'halo' region. Then, an update of boundary conditions is effected through the connectivity matrix and the identifiers of the coordinate systems in the neighboring blocks in order to establish the inter-block coupling. An 'outer iteration' consists of solving any one set of equations over all blocks and the associated exchange of data across block boundaries. This sweep is arranged as a *Block Jacobi* method. A sample multi-block mesh for a flow over a 2D two-element airfoil from [7] is illustrated in Figure 4. The multi-block code was originally designed to run on a shared-memory machine. Porting a multi-block code to a distributed-memory system will be discussed in the next section.

3. PORTING TO DISTRIBUTED-MEMORY MACHINE

The major change between the serial and parallel versions of urbanSTREAM lies in the index system in the DO loop, which is illustrated by the following partial FORTRAN listing of subroutine CALCU – a subroutine which solves the x-component of the velocity in the discretized Navier-Stokes equation:

Serial version:

```
DO K=2,NKM1
DO J=2,NJM1
DO I=2,NIM1

IF(iand(flag(i,j,k),C_O)/=0) THEN
  AB(I,J,K)=0.0
  AT(I,J,K)=0.0
  AS(I,J,K)=0.0
  AN(I,J,K)=0.0
  AW(I,J,K)=0.0
  AE(I,J,K)=0.0
  SU(I,J,K)=0.0
```

```
        SP(I,J,K)=0.0
        CYCLE
      END IF
•
•
•
      SU(I,J,K)=SU(I,J,K)
     1+DU1(I,J,K)*(PRESW-PRESE)
     1+DU2(I,J,K)*(PRESS-PRESN)
     1+DU3(I,J,K)*(PRESB-PREST)
     1+((AF+BT)*UO(I,J,K)-BT*UM(I,J,K))*VOL(I,J,K)
     1/DELT

      END DO
      END DO
      END DO
```

Parallel version:

```
      NB=1

      DO K=NKBEG(NB)+2,NKEND(NB)-2
      DO J=NJBEG(NB)+2,NJEND(NB)-2
      DO I=NIBEG(NB)+2,NIEND(NB)-2

      IF(iand(flag(i,j,k),C_O)/=0) THEN
        AB(I,J,K)=0.0
        AT(I,J,K)=0.0
        AS(I,J,K)=0.0
        AN(I,J,K)=0.0
        AW(I,J,K)=0.0
        AE(I,J,K)=0.0
        SU(I,J,K)=0.0
        SP(I,J,K)=0.0
        CYCLE
      END IF
•
•
•
      SU(I,J,K)=SU(I,J,K)
     1+DU1(I,J,K)*(PRESW-PRESE)
     1+DU2(I,J,K)*(PRESS-PRESN)
     1+DU3(I,J,K)*(PRESB-PREST)
     1+((AF+BT)*UO(I,J,K)-BT*UM(I,J,K))*VOL(I,J,K)
     1/DELT

      END DO
      END DO
      END DO
```

A data file containing coordinate information of a computational mesh generated by urbanGRID, called "grid_urbanSTREAM.dat", is read first into the grid partitioning program, called "gridgn_inflow90", which automatically partitions the single grid into NBLOC partitions of approximately equal sizes in order to achieve load-balancing (see Figure 5). Here, NBLOC = I_MAX × J_MAX × K_MAX, in which I_MAX, J_MAX and

K_MAX represent the number of partitions in the I-, J- and K-direction (in grid coordinates), respectively. The user needs to specify I_MAX, J_MAX and K_MAX in "partition.dat" in order to determine NBLOC, which is the same as the number of central processing units (CPUs) required to run urbanSTREAM-P. An example with I_MAX=2, J_MAX=2 and K_MAX=2 is illustrated in Figure 6 and one-eighth of the computational mesh is shown in Figure 7, in which the "halo data" regions are highlighted (see also Figure 8). Swapping of information in the halo data regions is achieved by calling the OVEL_Q subroutine. A partial FORTRAN listing of the subroutine OVEL_Q including message-passing interface MPI [2] routines on the eastern and western boundaries is given below:

```
C
C East
C
      IF(MCONEC(node+1,1).NE.0) THEN
C
C     packing message
C
      DO L=1,2*NY*NZ
      QSDE(L)=0.
      END DO
C
      L=0
      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      QSDE(L)=Q(NIEND(1)-2,J,K)
      END DO
      END DO

      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      QSDE(L)=Q(NIEND(1)-3,J,K)
      END DO
      END DO
C
      CALL MPI_SEND(QSDE,2*NY*NZ,MPI_REAL8,
     &MCONEC(node+1,1)-1,MSGPTR(node+1)+1,MPI_COMM_WORLD,ierr)
!
      CALL MPI_RECV(QREE,2*NY*NZ,MPI_REAL8,
     &MCONEC(node+1,1)-1,MSGPTR(MCONEC(node+1,1))+2,MPI_COMM_WORLD,
     &status,ierr)
C
C     unpacking message
C

      L=0
      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      Q(NIEND(1)-1,J,K)=QREE(L)
      END DO
      END DO
      DO K=NKBEG(1)+2,NKEND(1)-2
```

```
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      Q(NIEND(1)   ,J,K)=QREE(L)
      END DO
      END DO
C
      END IF
C
C West
C
      IF(MCONEC(node+1,2).NE.0) THEN
      CALL MPI_RECV(QREW,2*NY*NZ,MPI_REAL8,
     &MCONEC(node+1,2)-1,MSGPTR(MCONEC(node+1,2))+1,MPI_COMM_WORLD,
     &status,ierr)
C
C     unpacking message
C
      L=0
      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      Q(NIBEG(1)+1,J,K)=QREW(L)
      END DO
      END DO
      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      Q(NIBEG(1)   ,J,K)=QREW(L)
      END DO
      END DO
C
C     packing message
C
      DO L=1,2*NY*NZ
      QSDW(L)=0.
      END DO
C
      L=0
      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      QSDW(L)=Q(NIBEG(1)+2,J,K)
      END DO
      END DO
      DO K=NKBEG(1)+2,NKEND(1)-2
      DO J=NJBEG(1)+2,NJEND(1)-2
      L=L+1
      QSDW(L)=Q(NIBEG(1)+3,J,K)
      END DO
      END DO
C
      CALL MPI_SEND(QSDW,2*NY*NZ,MPI_REAL8,
     &MCONEC(node+1,2)-1,MSGPTR(node+1)+2,MPI_COMM_WORLD,ierr)
C
      END IF
```

There are two MPI routines used in OVEL_Q, namely MPI_SEND and MPI_RECV, which send messages to and receive messages from the neighboring blocks, respectively (see Figure 9). The index of the neighboring block (or, equivalently, the node number in the parallel system) is identified by the connectivity matrix MCONEC as explained in Section 2. The index system used for a block in question and four neighboring halo data regions in a 2-D environment is presented in Figure 10, to facilitate a better understanding of the above sample code. Note that MPI_SEND is "blocking". This means that it does not return until the message data have been stored away, so that the sender is free to access and overwrite the send buffer. Similarly, MPI_RECV is also blocking. It returns only after the receive buffer contains the newly received message. Message buffering decouples the send and receive operations. A blocking send might complete as soon as the message was buffered, even if no matching receive has been recorded by the receiver. On the other hand, message buffering can be expensive, as it entails additional memory-to-memory copying, and it requires the allocation of memory. In an ill-constructed program, blocking may lead to a "deadlock" situation, where all processes are blocked (deadlocked), and no progress occurs. Such programs may complete when sufficient buffer space is available, but will fail on systems that use less buffering, or when message sizes are increased. Since any system will run out of buffer resources as message sizes are increased, a "safe programming style" is adopted in the present implementation as illustrated in the above sample code, which succeeds even if no buffer space is available and, therefore, is safe and will always complete correctly.

When the line-by-line TDMA (Tri-Diagonal Matrix Algorithm) [8] is used in the LISOLV subroutine, OVEL_P is inserted after LISOLV as exemplified in a partial list of FORTRAN code in the CALCU subroutine below:

```
      DO N=1,NSWPU

      CALL LISOLV(U)
      CALL MPI_BARRIER(MPI_COMM_WORLD,ierr)
      CALL OVEL_Q(U)
      END DO
C
      CALL OVEL_Q(AP)
C
      DO K=NKBEG(NB)+1,NKEND(NB)-1
      DO J=NJBEG(NB)+1,NJEND(NB)-1
      DO I=NIBEG(NB)+1,NIEND(NB)-1

      DU1(I,J,K)=XX(I,J,K)/(AP(I,J,K)+TINY)
      DU2(I,J,K)=YX(I,J,K)/(AP(I,J,K)+TINY)
      DU3(I,J,K)=ZX(I,J,K)/(AP(I,J,K)+TINY)

      END DO
      END DO
      END DO
```

Note also that it is important to swap a_P between halo data regions, as described in Section 2, in order to ensure that mass fluxes at block boundaries, evaluated using Rhie-Chow interpolation [4], are implemented correctly.

4. APPLICATION TO AN ARRAY OF OBSTACLES

The geometry of the obstacle array is exhibited in Figure 11. Meinders and Hanjalic [9] made detailed measurements of the mean flow and turbulence energy using a two-component laser Doppler anemometer within this array of obstacles. To simulate the flow in this array, we used a computational domain consisting of a sub-channel unit of $4H\times 3.4H\times 4H$ as shown in Figure 11. Calculations were performed on a mesh of 85×45×45 grid lines in x-, y- and z-directions, respectively, using the standard k-ε turbulence model. These directions correspond, respectively, to the streamwise, vertical, and spanwise directions in the plane channel. The turbulence kinetic energy (TKE) contours, which are continuous across block boundaries, verify that the message-passing routines (MPI_SEND and MPI_RECV) in OVEL_P are implemented correctly. Figure 12 shows a comparison between predicted and measured streamwise velocity and TKE profiles at two stations, x/H = -0.3 and 0.3. Agreement in terms of streamwise velocity is generally fairly good. However, URANS tends to under-predict TKE within the street canyon. It is also observed in Figure 12 that sensitivities of the solutions to two different time-stepping schemes are very weak. The parallel efficiency, measured on the Nexus and Australis computer systems using up to 32 CPUs on Westgrid (www.westgrid.ca), is exhibited in Figure 13. Westgrid is a grid-enabled system for high performance computing across Western Canada. As seen, the parallel efficiency relative to 2 CPUs, defined as

$$\text{Parallel Efficiency}=\frac{T_{2CPUs}\ \text{per iteration}}{(n/2)T_{nCPUs}\ \text{per iteration}} \tag{3}$$

where T stands for CPU time, drops to 81% as 32 CPUs are used.

5. APPLICATION TO JOINT URBAN 2003

A major field experiment, Joint Urban 2003 (JU2003) [10], was conducted in Oklahoma City from June 28 to July 31, 2003 to collect meteorological and tracer data sets for evaluating dispersion models in urban areas. Various meteorological instrumentation and tracer samplers were installed at various locations throughout and around the city in order to track the air movement of safe, non-toxic tracer gases. The purpose of this study is to provide a better understanding of pollutant dispersion in an urban environment, and the data can be used to develop and validate urban models for flow and dispersion. Further details of the experiment can be found at https://ju2003-dpg.dpg.army.mil/.

The computational domain consists of an "inner region", in which all buildings are explicitly resolved as shown in Figure 14, and an "outer region" in which the effect of an aggregation of groups of buildings are treated as a porous barrier [11] with a normalized drag coefficient $C_d \equiv \hat{C}_d \hat{A} \Lambda_{ref} = 100$ (where $\Lambda_{ref} = 645\,\text{m}$ is the reference length scale). A mesh of 99×139×69 grid lines was used to cover the entire computational domain (inner

plus outer regions), inside which a mesh of 55×100×69 grid lines was used for the inner region. The robustness of urbanSTREAM-P is demonstrated by performing calculations using power-law velocity profiles as inflow conditions for the south-east, south-west, north-east and north-west wind directions as shown in Figure 15. The parallel efficiency, defined as

$$\text{Parallel Efficiency} = \frac{T_{1\text{CPUs}} \text{ per iteration}}{(n)T_{n\text{CPUs}} \text{ per iteration}} \tag{4}$$

which was measured on the Narwhal computer system on Sharcnet (www.sharcnet.ca) using up to 32 CPUs, for the south-west wind direction is exhibited in Figure 16. The parallel efficiency drops to 48% when 32 CPUs were used, suggesting that the "safe" MPI implementation described in Section 3 will need further improvement.

To demonstrate consistency, vertical profiles of mean wind speed and wind direction at 35.46474° N and 97.51822° W using inflow conditions interpolated from the early results of the urban GEM LAM model [12] for both serial and parallel versions of urbanSTREAM (urbanSTREAM-S and urbanSTREAM-P, respectively) are shown in Figure 17. It is encouraging to see from Figure 17 that identical results are obtained using the serial and parallel versions of urbanSTREAM. Since then, an improved inflow condition from the urban GEM LAM model has been provided by the Canadian Meteorological Center (CMC) and is exhibited in Figure 18, in which a comparison with measurements for IOP-9 at 06:30 UTC on 28 July 2003 is made at 35.45295° N and 97.52534° W, which we refer to as the PNNL (Pacific Northwest National Laboratory) location hereafter. The PNNL location is approximately 2 km south of the central business district of Oklahoma City. Note that the mean wind speed is reasonably well predicted except near the ground ($z \leq 40$ m), where the mean wind and turbulence were obtained using an analytical 1-D flow model [13] to extrapolate the flow quantities predicted by urban GEM LAM above the canopy to locations within the canopy. The wind direction, in the range of vertical heights $75 \leq z \leq 200$ m, is incorrect. In particular, measurements suggest that the mean wind direction is ≈176°, whereas the urban GEM LAM model predicts a wind direction of ≈183°. The effect of this minor difference in the mean wind direction on the dispersion predictions will be discussed later.

Figures 19 and 20 compare the predicted vertical profiles of the mean horizontal wind speed and wind direction with associated measured values obtained using minisodars at 35.46474° N and 97.51822° W (in the Botanical Gardens), and at 35.47866° N and 97.51707° W (near 10th Street and Harvey Avenue), respectively. The minisodars were deployed by Argonne National Laboratory (ANL). As can be seen, the horizontal wind speeds in both figures predicted by RANS agree reasonably well with measurements. However, the predicted wind direction (≈180°) for $z \leq 100$ m in Figure 19 is erroneous compared to the measured values, which are ≈130° to 180°.

The flow field statistics predicted using RANS were next used to "drive" an urban dispersion model, urbanEU (see, e.g., [14] for details). The simulations were conducted for the second continuous 30-min release of SF_6 in IOP-9, which occurred during the

period from 06:00-06:30 UTC on 28 July 2003. The dissemination point was located on the south side of Park Avenue at 35.46871° N and 97.51556° W, with a release height of 1.9 m. The constant gas release rate for this experiment was 2.0 g s^{-1}. Figure 21 displays various isopleths (on a logarithmic scale) of the predicted mean concentration field close to ground level obtained using urbanEU.

Figures 21-24 compare predictions of the mean concentrations [in parts-per-trillion by volume (pptv)] obtained with (1) prescribed wind profiles according to the power laws at the PNNL location and a wind direction of ≈180°, and (2) wind profiles and wind directions interpolated from the urban GEM LAM model at various sampling locations in Oklahoma City. The experimental data shown here is for a 30-min averaging time. Generally speaking, the predictions for the mean concentration displayed in Figure 24, identified as "urbanEU (with PNNL)", for sampling locations along 6th Street and along the 1-km sampling arc were quite good at or near the mean plume centerline, with predictions within a factor of two of the observed concentration. In contrast, results identified as "urbanEU (with GEM/LAM)" in Figure 24 suggested that the predicted mean plume centerline was too far east, which was consistent with the predicted wind direction, which averaged approximately 25° greater than the measured values for $0 \leq z \leq 100$ m (i.e., the predicted wind direction was too far east), as shown in Figure 19. This was likely caused by the discrepancy in the wind direction between GEM LAM predictions and the experimental data observed at the inflow plane as seen in Figure 18 earlier.

6. CONCLUSIONS

Details of the parallelization of urbanSTREAM and the testing of its performance on various parallel computer systems have been addressed in this report. The major steps involved in converting the serial version of urbanSTREAM into its parallel version – termed urbanSTREAM-P, are (1) the employment of the "domain-decomposition" strategy (or, equivalently, the multi-block approach) so that block connectivity information and "halo data" regions can be constructed, and (2) explicit passing of messages stored in the halo data regions among different nodes using the MPI library to establish a coupling of the solutions in each sub-domain, which collectively form the original, entire computational domain. In addition, changes to the index system in the DO loop are also necessary, which will facilitate the future implementation of the fully-coupled multigrid (MG) scheme and the local-grid-refinement (LGR) method to be addressed later.

One-way coupling between urbanSTREAM and the urban GEM LAM model is implemented in the system and validated against the JU2003 database for the IOP9 test problem. The flow data, including 3-D flow field and turbulence kinetic energy (TKE) profiles, provided by urban GEM LAM, are first interpolated by urbanGRID and used as inflow conditions to "drive" urbanSTREAM. urbanSTREAM, in turn, generates time-averaged velocity and turbulence fields to "drive" urbanEU to generate concentration fields. The test calculations permit the following conclusions to be drawn and recommendations to be made.

1. The sensitivity of concentration predictions to the inflow condition, particularly in terms of a wind direction that is different by a few degrees, was observed for the IOP9 test problem.

2. It is expected that better concentration predictions could be obtained if urbanEU were to be combined with urbanSTREAM (viz., adding an advection-diffusion equation for concentration in urbanSTREAM) and executed in a true (instead of pseudo) *unsteady* RANS mode with inflow conditions from the GEM LAM database being updated every 30 minutes.

3. The robustness and consistency of urbanSTREAM-P have been demonstrated by testing the code with four different wind direction scenarios, namely north-east, north-west, south-east and south-west, and comparing the results obtained on the same grid when using both the serial and parallel versions of urbanSTREAM.

4. The wall-clock time required for urbanSTEAM-P is significantly reduced when more CPUs are used. Hence, the parallel efficiency achieved in the present study still requires further improvement. For 32 CPUs, an approximate parallel efficiency of 50% was achieved, which is likely due to the employment of a "safe" programming style based on "blocking" send and receive MPI routines. It is recommended that "non-blocking" send and receive MPI routines be implemented, although care must be taken to avoid "deadlock", particularly for systems in which buffer space is limited.

5. To further improve the computational efficiency, in particular when thermal stratification effects are important and the coupling between velocity and temperature fields is strong, the fully-coupled multigrid (MG) method [15] as illustrated in Figure 25, will need to be implemented. Here, the primary task is to compute corrections of variables for a system of transport equations, including momentum, pressure-correction, energy and turbulence equations, instead of correction of a variable for each transport equation *separately* on a sequence of meshes of different sizes. The main challenge behind applying MG for turbulent flows is that the TKE can potentially be "negative" (or, non-realizable) at the prolongation (coarse-to-fine grid interpolation) stage during MG cycles.

6. urbanGRID is designed for use by a standard structured-grid flow solver, such as urbanSTREAM. In the case of multiple source scenarios (viz., CBRN agents released in several locations throughout a city), the capability of local-grid-refinement (LGR), as exemplified in Figure 26, can be beneficial from the point of view of both computational efficiency and solution accuracy. LGR allows sufficient grid resolution to be provided in regions of great interest, such as clusters of buildings surrounding the source. In addition, the computational cost, which is closely related to the total number of grid points, is significantly reduced compared to the case where an equally-fine grid is used for the entire solution domain. However, parallelizing a LGR method can be challenging. A (close to) perfect load balancing is deemed necessary in order to minimize the total CPU

time required on any parallel (distributed-memory or shared-memory) systems. It is recommended that the Hilbert's space-filling-curve (HSFC) approach be used as demonstrated in [16].

REFERENCES

1. E. Yee, F.S. Lien and H. Ji (2007), "Technical Description of Urban Microscale Modeling System: Component 1 of CRTI Project 02-0093RD", Defence R&D Canada – Suffield, Technical Report DRDC Suffield TR 2007-067.
2. MPI. http://www-unix.mcs.anl.gov/mpi/.
3. F.S. Lien, W.L. Chen and M.A. Leschziner (1996), "A Multi-Block Implementation of Non-Orthogonal Collocated Finite-Volume Algorithm for Complex Turbulent Flows", *Int. J. Num. Meth. Fluids*, **23**, 567-588.
4. C.M. Rhie and W.L. Chow (1983), "Numerical Study of the Turbulent Flow past an Airfoil with Trailing Edge Separation", *AIAA J.*, **21**, 1525-1532.
5. S.V. Patankar (1980), Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, New York.
6. H.L. Stone (1968), "Iterative Solution of Implicit Approximations of Multi-dimensional Partial Differential Equations", *SIAM J. Numer. Anal.*, **5**, 530-558.
7. F.S. Lien, W.L. Chen and M.A. Leschziner (1995), "Computational Modelling of High-Lift Aerofoils with Turbulence-Transport Models", *Proc. CEAS European Forum High Lift & Separation Control*, Bath, UK, 10.1-10.14.
8. L.H. Thomas (1949), "Elliptic Problems in Linear Difference Equations over a Network", Watson Sci. Comput. Lab. Report, Columbia University, New York.
9. E.R. Meinders and K. Hanjalic (1999), "Vortex Structure and Heat Transfer in Turbulent Flow over a Wall-mounted Matrix of Cubes", *Int. J. Heat Fluid Flow*, **20**, 255-267.
10. K.J. Allwine and J.E. Flaherty (2006), "Joint Urban 2003: Study Overview and Instrument Locations", Pacific Northwest National Laboratory Technical Report PNNL-15967.
11. F.S. Lien, E. Yee and J.D. Wilson (2005), "Numerical Modelling of the Turbulent Flow Developing within and over a 3-D Building Array, Part II: Mathematical Formulation for a Distributed Drag Force Approach", *Boundary-Layer Meteorology*, **114**, 245-285.
12. J. Cote, S. Gravel, A, Methot, A. Patoine, M. Roch and A. Staniforth, "The Operational CMC-MRB Global Environment Multiscale (GEM) Model, Part I: Design Considerations and Formulation", *Monthly Weather Review*, **126**, 1373-1395.
13. W.J. Massman and J.C. Weil (1999), "An Analytical One-dimensional Second-order Closure Model of Turbulence Statistics and the Lagrangian Time Scale within and above Plant Canopies of Arbitrary Structure", *Boundary-Layer Meteorology*, **91**, 81-107.
14. K.J. Hsieh, F.S. Lien and E. Yee (2007), "Numerical Modeling of Passive Scalar Dispersion in an Urban Canopy Layer", to appear in *Journal of Wind Engineering and Industrial Aerodynamics*.
15. F.S. Lien and M.A. Leschziner (1994), "Multigrid Acceleration for Turbulent Flow with a Non-Orthogonal, Collocated Scheme, *Comp. Meth. Appl. Mech. Eng.*, **118**, 351-371.
16. H. Ji, F.S. Lien and E. Yee (2006), "Parallel Adaptive Mesh Refinement Combined With Multigrid for a Poisson Equation", *Proc. 14th Annual Conference of the Computational Fluid Dynamics Society of Canada*, July 16-18, Kingston, Canada.

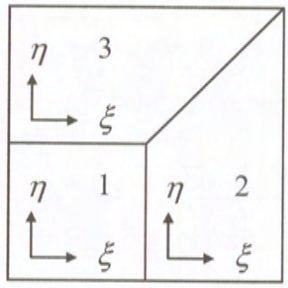


Figure 1. Multi-block arrangement with different local coordinate systems.

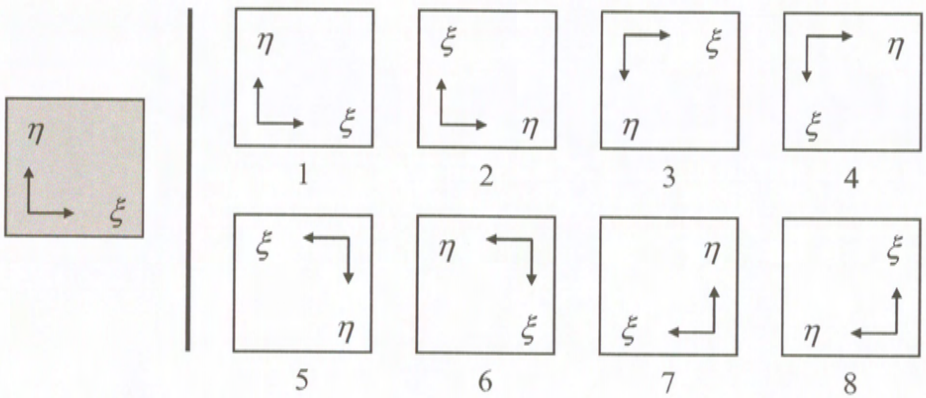


Figure 2. All possible permutations of local coordinate systems in blocks adjacent to any reference block.

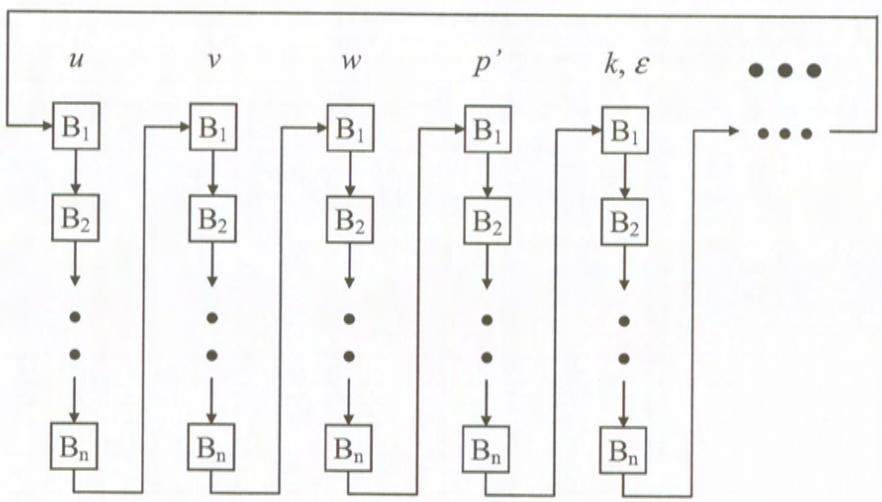


Figure 3. Solution sequence of SIMPLE algorithm within a multi-block scheme.

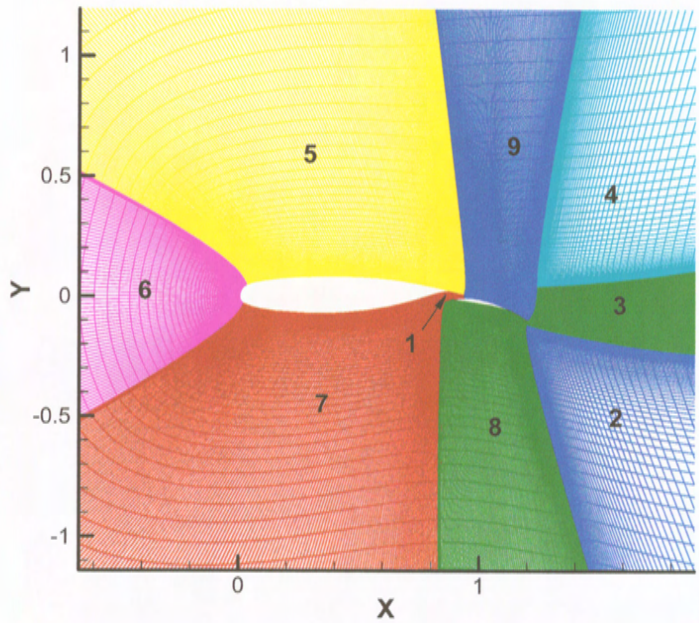


Figure 4. A multi-block mesh for a 2D two-element airfoil [7].

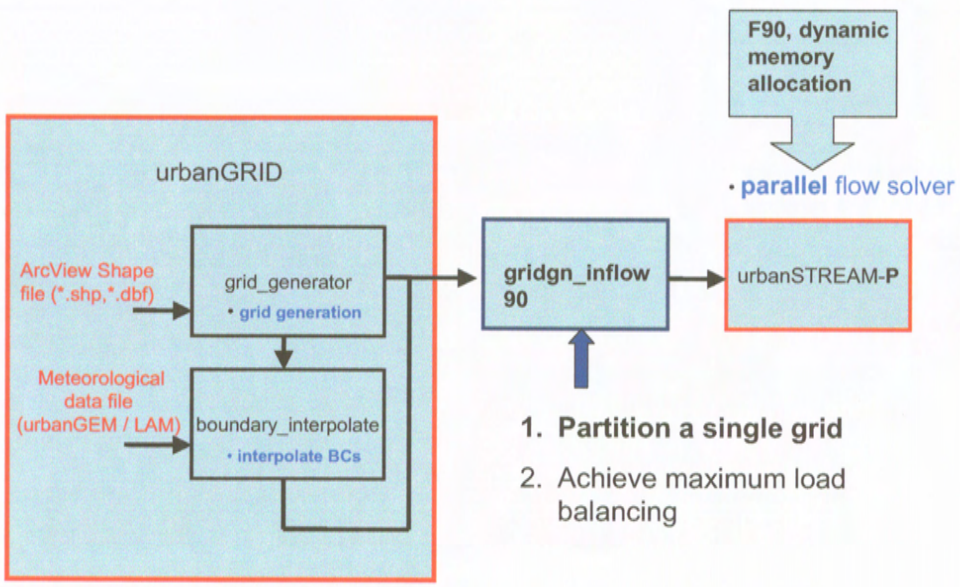


Figure 5. Interfacing urbanGRID and urbanSTREAM-P.

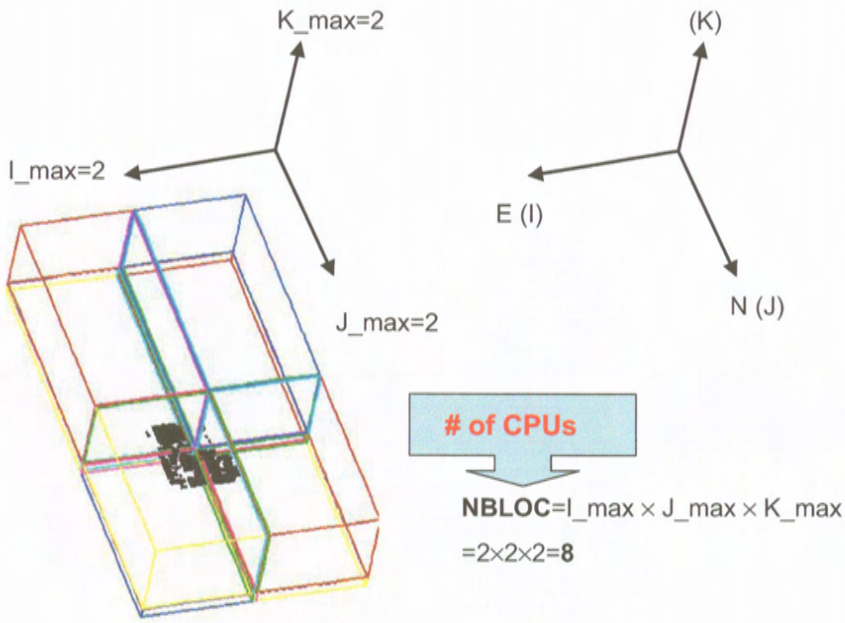


Figure 6. Domain decomposition of a solution domain for 8 CPUs.

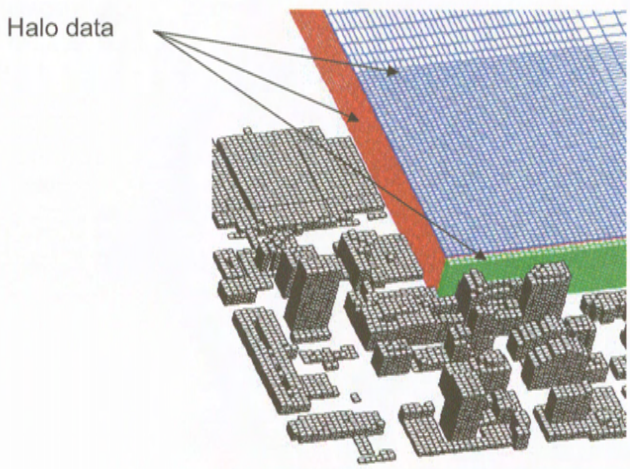


Figure 7. One-eighth of the computational mesh, including the "halo data" regions.

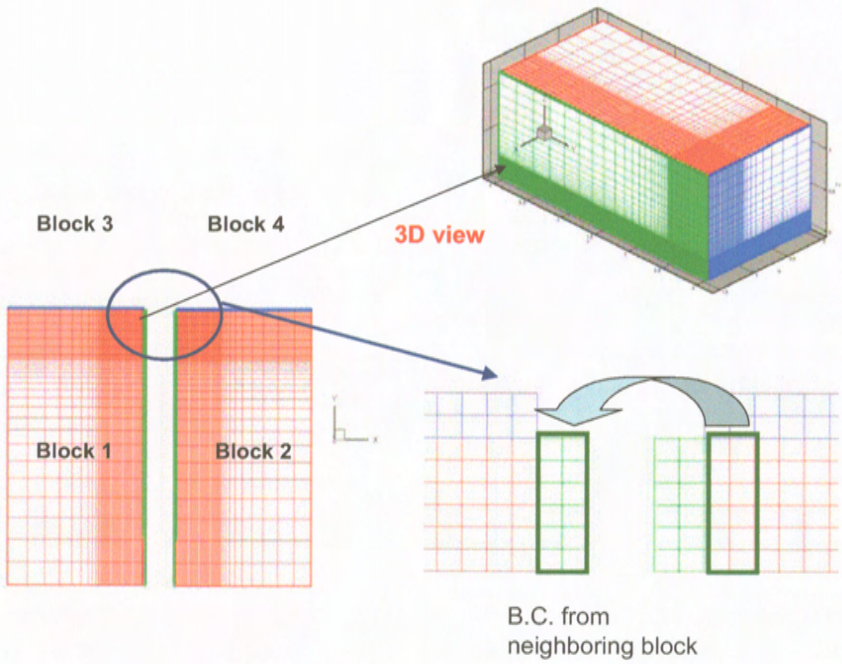


Figure 8. Illustration of data swapping between two adjacent halo data regions.

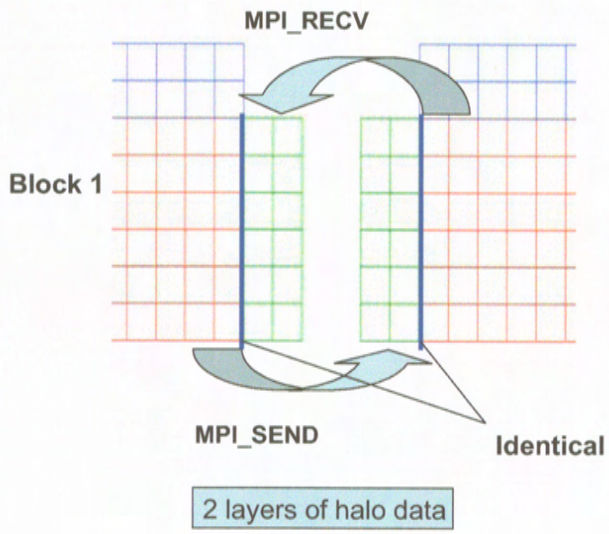


Figure 9 Illustration of data swapping by using MPI_SEND and MPI_RECV routines in halo data regions.

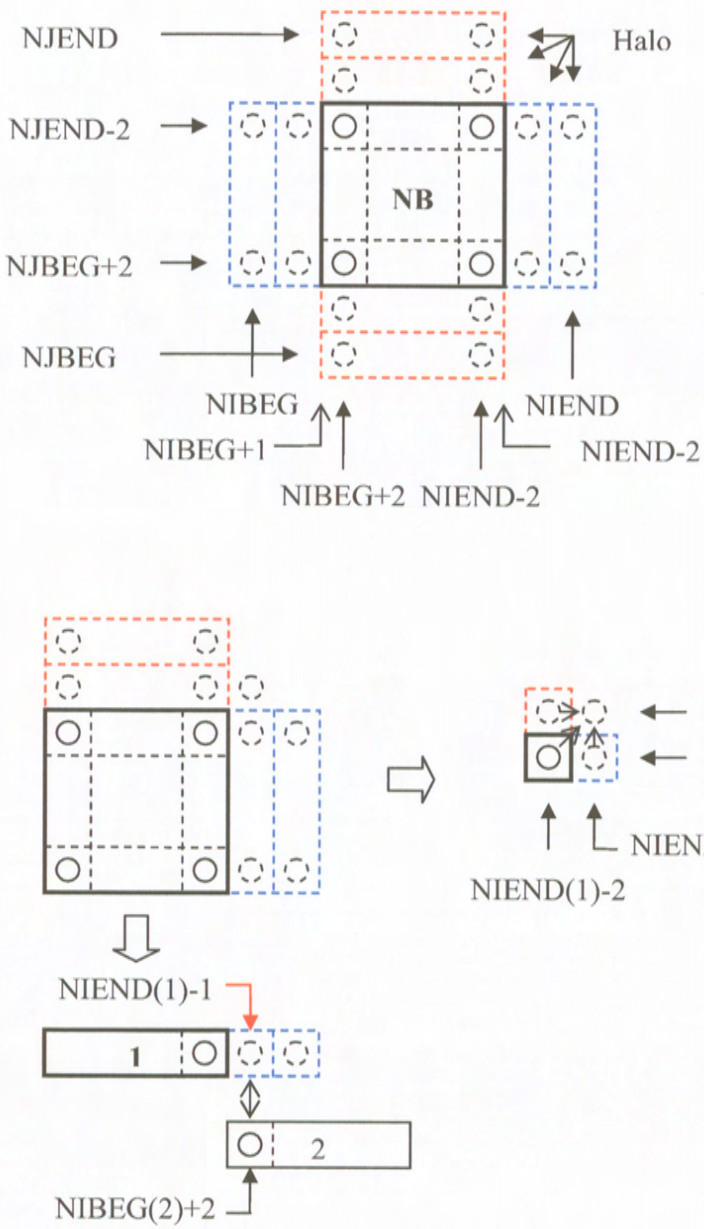


Figure 10 Index system used for the block in question and four neighboring blocks in a 2-D environment.

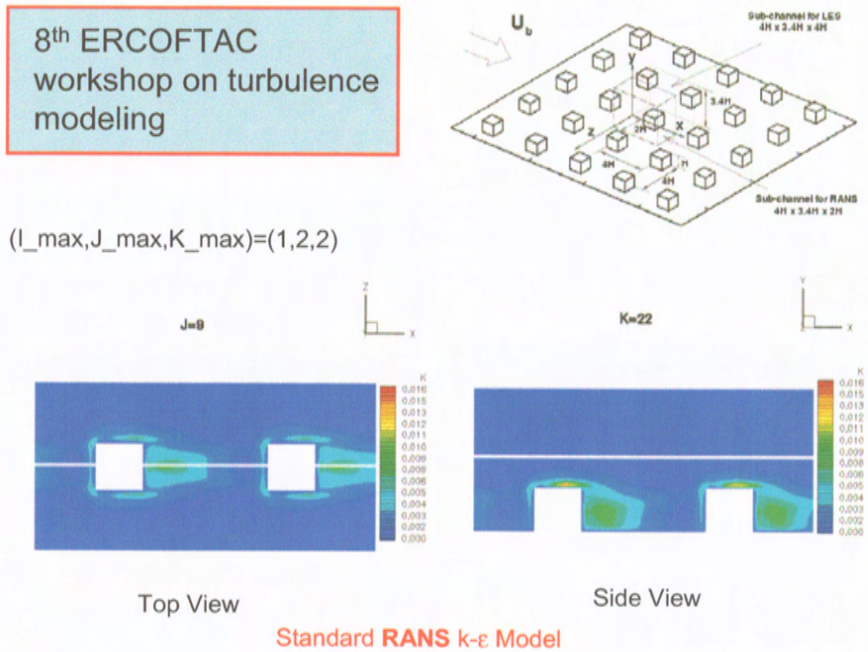


Figure 11. Flow over a regular and aligned array of obstacles obtained with URANS on a 4-CPU parallel system.

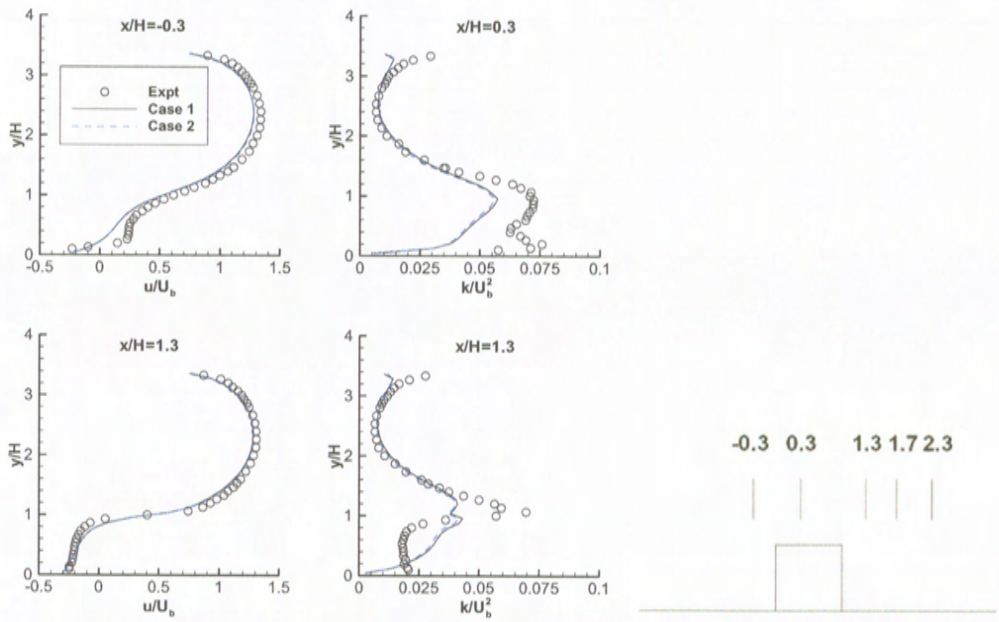


Figure 12. Profiles of streamwise velocity and turbulence kinetic energy (TKE) for a flow over a regular and aligned array of obstacles using URANS. Case 1: 2nd-order time-stepping; Case 2: 1st-order time stepping.

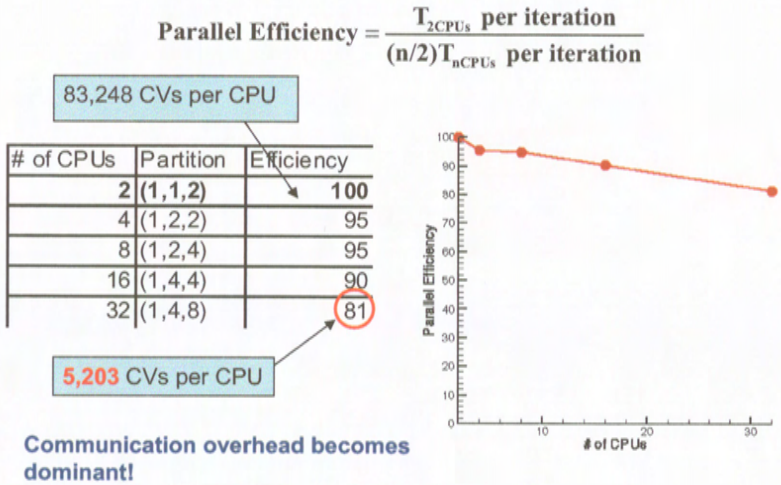


Figure 13. Parallel efficiency measured on the Nexus and Australis computing systems on Westgrid (www.westgrid.ca) for a flow over a regular and aligned array of obstacles using URANS.

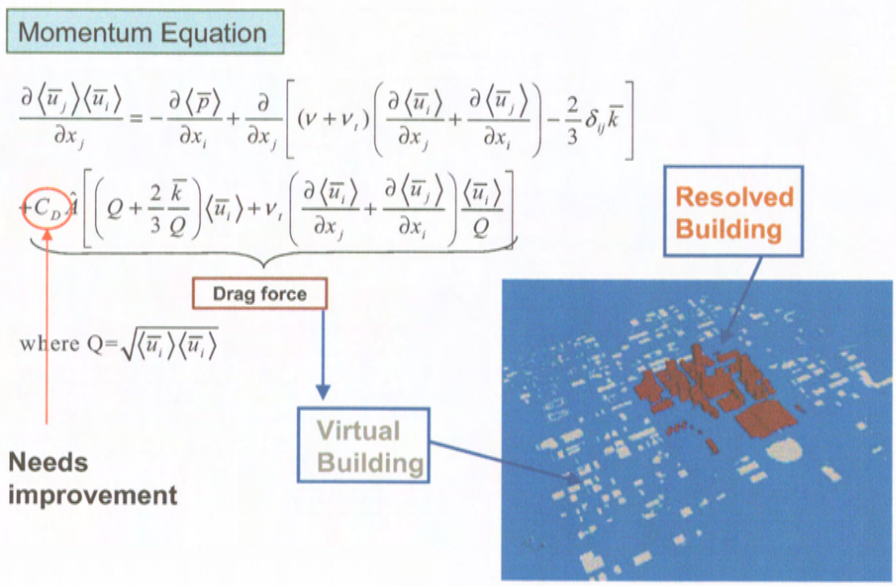


Figure 14. Illustration of the present drag-force approach applied to the JU2003 test problem.

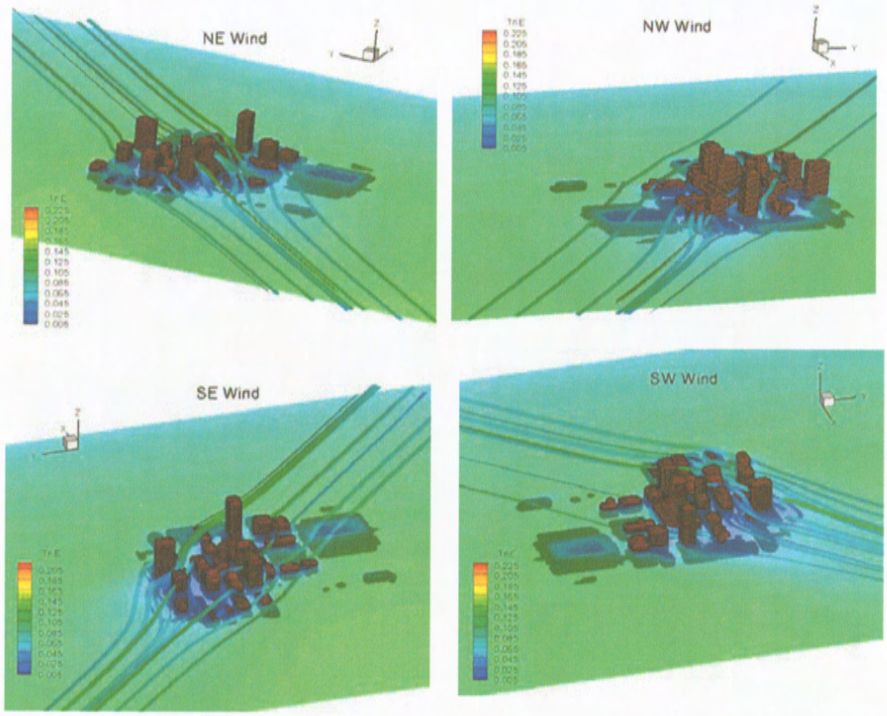


Figure 15. Flow visualization obtained with urbanSTREAM-P using power-law velocity profiles as inflow conditions for south-east, south-west, north-east and north-west wind directions in Oklahoma City.

$$\text{Parallel Efficiency} = \frac{T_{1CPU} \text{ per iteration}}{(n)T_{nCPU} \text{ per iteration}}$$

# of CPUs	Partition	Efficiency
1	(1x1x1)	100
2	(2x1x1)	87
4	(2x2x1)	81
8	(2x2x2)	79
16	(4x2x2)	63
32	(4x4x2)	48

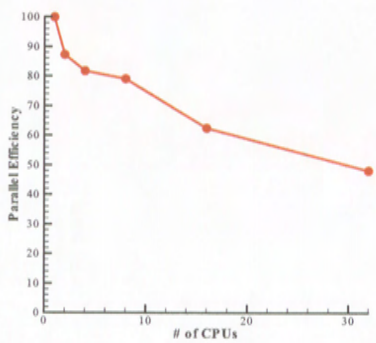


Figure 16. Parallel efficiency measured on the Narwhal computing system on Sharcnet (www.sharcnet.ca) for a flow over Oklahoma City for south-west wind direction using URANS.

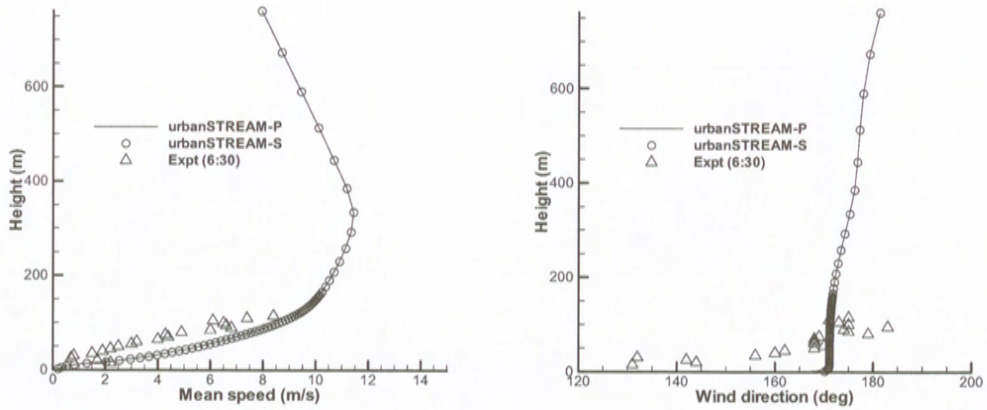


Figure 17. Vertical Profiles of mean wind speed and wind direction at 35.46474° N and 97.51822° W using inflow conditions interpolated from the early results of the urban GEM LAM model for both serial and parallel versions of urbanSTREAM.

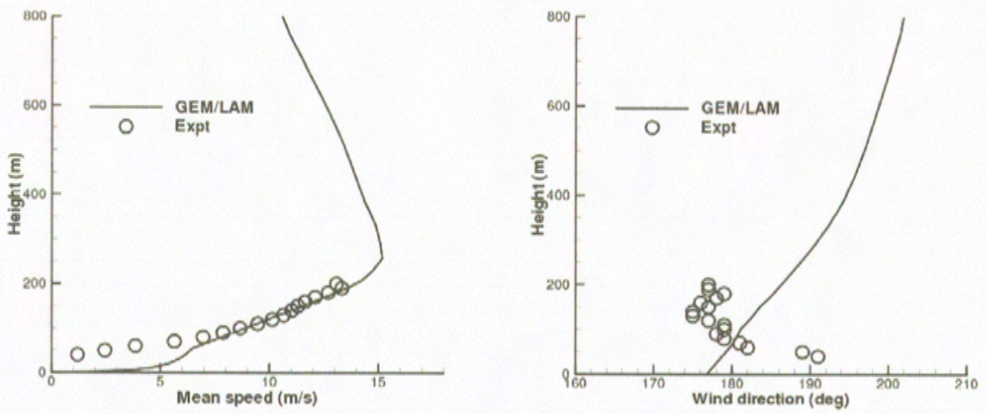


Figure 18. Comparison of vertical profiles of mean wind speed and wind direction interpolated from GEM LAM with experimental measurements at 35.45295° N and 97.52534° W, which is at 2 km south of the central business district of Oklahoma City.

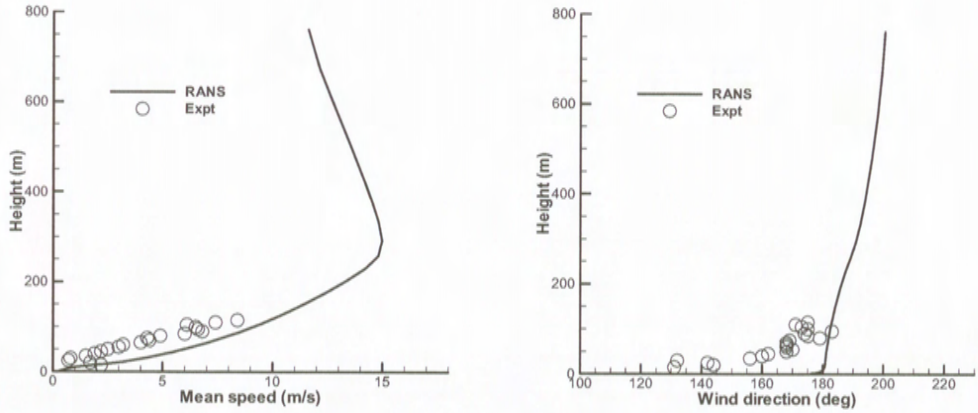


Figure 19. Comparison of vertical profiles of mean wind speed and wind direction interpolated from GEM LAM with experimental measurements at 35.46474° N and 97.51822° W.

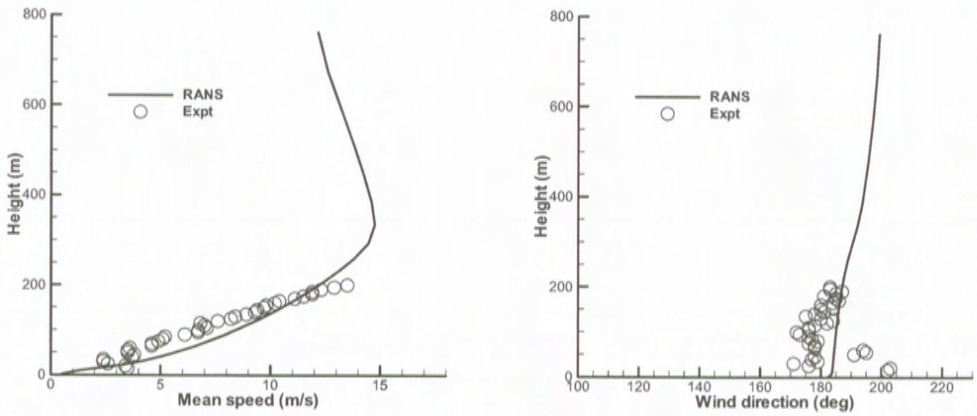


Figure 20. Comparison of vertical profiles of mean wind speed and wind direction interpolated from GEM LAM with experimental measurements at 35.47866° N and 97.51707° W.

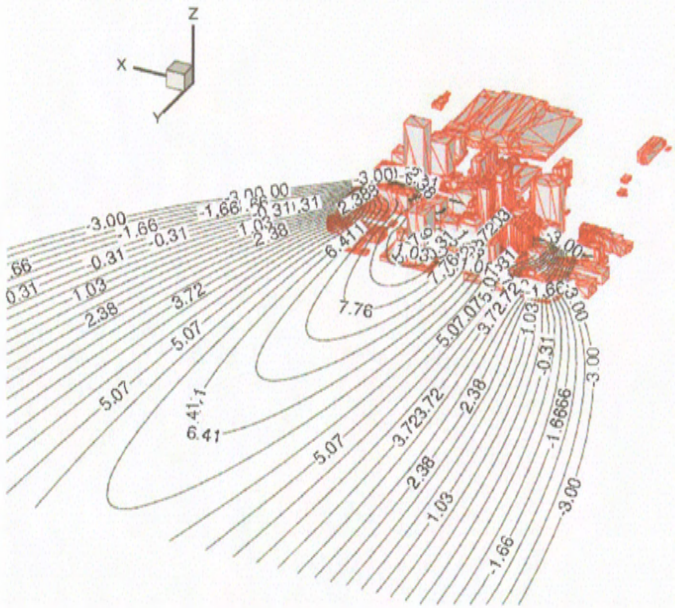


Figure 21. Mean plume concentration isopleths (logarithmic scale) at ground level for tracer release on south side of Park Avenue in Oklahoma City.

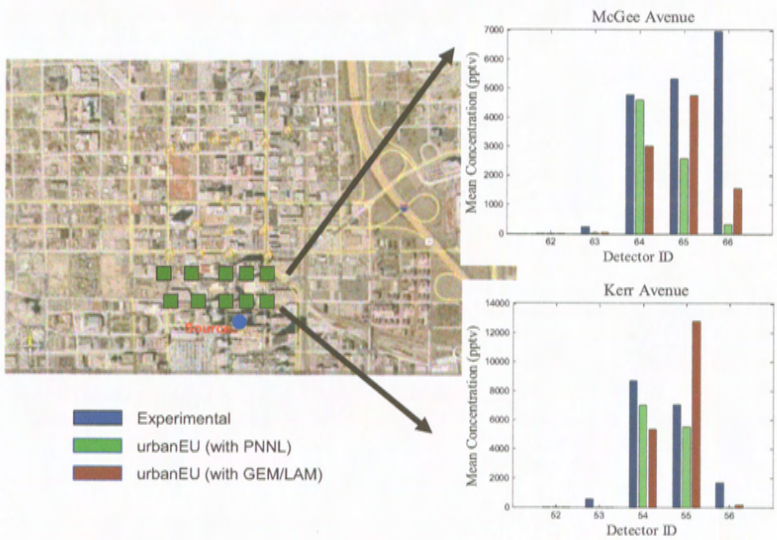


Figure 22. Comparison between predicted mean concentrations obtained from urbanEU with flow fields provided by urbanSTREAM-P in standalone (PNNL) and coupled (GEM/LAM) modes, and experimental measurements obtained with detectors along Kerr Avenue and McGee Avenue.

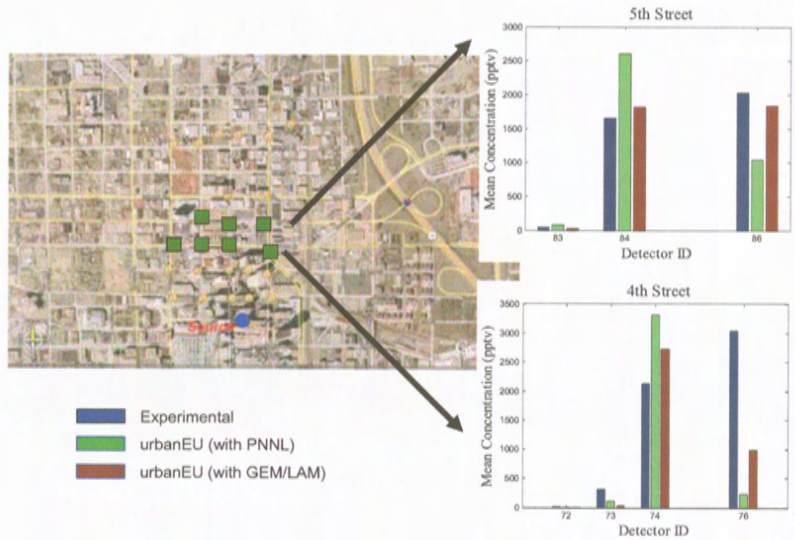


Figure 23. Comparison between predicted mean concentrations obtained from urbanEU with flow fields provided by urbanSTREAM-P in standalone (PNNL) and coupled (GEM/LAM) modes, and experimental measurements obtained with detectors along 4th Street and 5th Street.

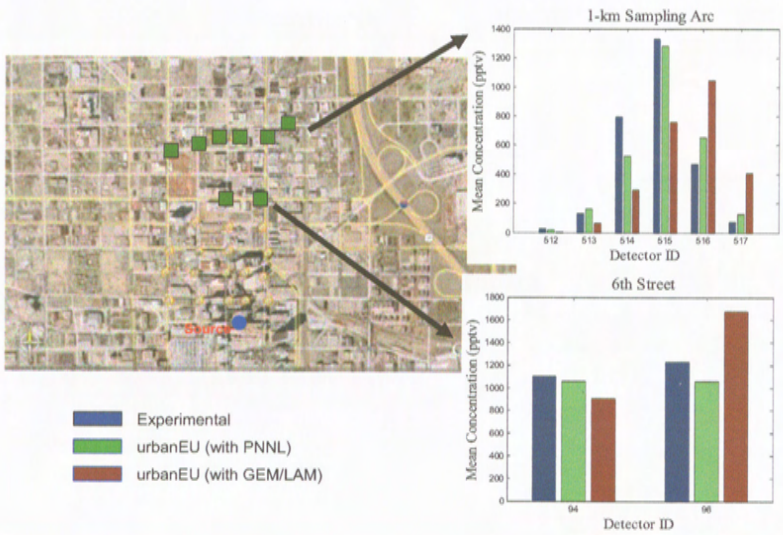


Figure 24. Comparison between predicted mean concentrations obtained from urbanEU with flow fields provided by urbanSTREAM-P in standalone (PNNL) and coupled (GEM/LAM) modes, and experimental measurements obtained with detectors along 6th Street and the 1-km sampling arc.

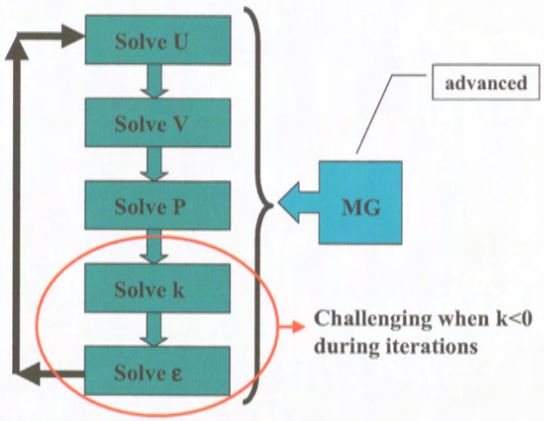


Figure 25. Flow chart of fully-coupled multigrid method for 2-D turbulent flows.

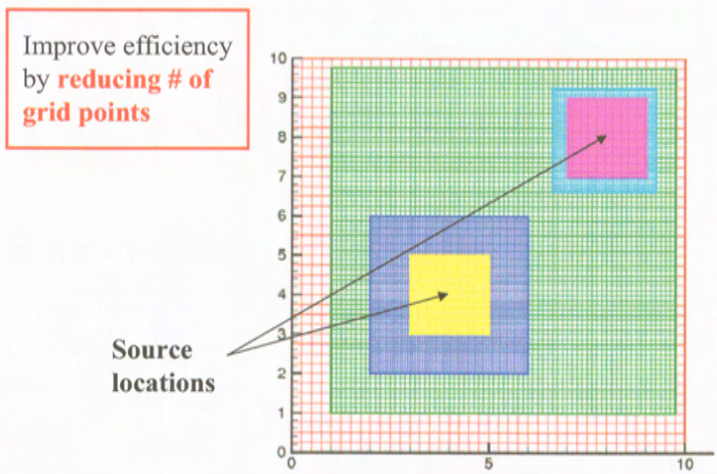


Figure 26. Mesh generation for local-grid-refinement with application to multiple source scenarios.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for who the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in Section 8.)

Waterloo CFD Engineering Consulting, Inc.
200 University Avenue West
Waterloo, ON N2L 3G1

2. SECURITY CLASSIFICATION
(overall security classification of the document, including special warning terms if applicable)

Unclassified

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title).

Parallelization of an Urban Microscale Flow Model (urbanSTREAM): Component 1 of CRTI Project 02-0093RD

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

Lien, F.-S. and Hsieh, K.-J.

5. DATE OF PUBLICATION (month and year of publication of document)

December 2007

6a. NO. OF PAGES (total containing information, include Annexes, Appendices, etc) 31

6b. NO. OF REFS (total cited in document) 16

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Final contract report

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

Defence R&D Canada – Suffield, PO Box 4000, Station Main, Medicine Hat, AB T1A 8K6

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

CRTI-02-0093RD

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

W7702-03-R966

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

DRDC Suffield CR 2008-024

10b. OTHER DOCUMENT NOs. (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

WatCFD Report WD001/07

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(x) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally corresponded to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected).

No limitation on document announcement

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C) or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

The provision of an efficient and accurate Computational Fluid Dynamics (CFD) solver for simulating flows within an urban complex on the microscale (with characteristic length scale extending to about 1-2 km) is required to support the development of an advanced atmospheric transport and diffusion multiscale modeling system for hazard assessment of chemical, biological, radiological or nuclear (CBRN) agents released within cities. The ultimate goal of this supporting effort is to construct a virtual test facility (VTF) that could eventually be used to develop guidelines and procedures for operating in the urban complex after a CBRN incident.

Details of the numerical algorithms used for the "stand-alone" microscale CFD solver urbanSTREAM executing on serial computers have been described in the report entitled "Technical Description of Urban Microscale Modeling System: Component 1 of CRTI Project 02-0093RD" by E. Yee, F.-S. Lien and H. Ji (DRDC Suffield TR 2007-067). The objectives of this report are twofold. The first objective is to provide a description of the coupling of urbanSTREAM with an "urbanized" meso-gamma scale flow model (Global Environmental Multi-scale Local Area Model, or GEM LAM, developed by Canadian Meteorological Centre). A "one-way interaction" scheme is adopted, which allows for the matching of velocity and turbulence field in any overlap region in a physically and mathematically consistent manner that preserves the physical conservation laws, mutually satisfies mathematical boundary conditions, and preserves numerical accuracy. The second objective is to provide implementation details about how urbanSTREAM is parallelized for use on PC clusters and the IBM massively parallel supercomputing platform to give a problem-solving environment for full 3-D parallel simulation. Validation of this fully-coupled mesoscale-to-microscale system is achieved by detailed comparison of model predictions with comprehensive, high-quality full-scale urban measurements in a real cityscape — the Joint Urban 2003 (JU2003) conducted in Oklahoma City in the US.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifies, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

Computational Fluid Dynamics
MPI parallelization
Urban Flow simulation

Defence R&D Canada

Canada's Leader in Defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

www.drdc-rddc.gc.ca